Amendments to the Specification:

Please replace the paragraph at p. 2, lines 17-28 with the following amended paragraph:

According to the invention, in a digital information processing system wherein a model of a finite state machine (FSM) receiving a plurality of FSM inputs and producing a plurality of FSM outputs is represented by a reduced-state trellis and wherein the FSM inputs are defined on a base ~~closed~~ set of symbols, a novel method is presented for updating soft decision information on the FSM inputs into higher confidence information whereby (1) the soft decision information is inputted in a first index set, (2) a forward recursion is processed on the input soft decision information based on the reduced-state trellis representation to produce forward state metrics, (3) a backward recursion is processed on the input soft decision information based on the reduced-state trellis representation to produce backward state metrics, wherein the backward recursion is independent of the forward recursion and (4) the forward state metrics and the backward state metrics are operated on to produce the higher confidence information.

Please replace the paragraph at p. 4, lines 5-22 with the following amended paragraph:

Figure 1 shows a digital communication system 10 employing a trellis coded modulation (TCM) encoder 14. Such a TCM encoder 14 is just one of the many processes that can be modeled as a finite state machine (FSM). Other such processes include, but are not limited to, encoders based on Turbo codes or other codes, near-capacity multi-user detection, near-optimal two-dimensional data detection, fading channels, and inter-symbol interference (ISI) channels. Referring to Figure 1, the digital communication system 10 includes a data source 12 providing symbols defined on a particular ~~closed~~ set of symbols. For example, if a binary ~~closed~~ set of symbols is used, the symbols are selected from {0, 1}. The symbols from the data source 12 are transferred to the TCM encoder 14, which converts the symbols into encoded symbols according to the structure of the TCM encoder 14. The encoded symbols are then sent through a channel 16, which can add noise, such as additive white Gaussian noise (AWGN), and distortions to the encoded symbol to produce an observed signal. Soft information relating to the observed signal is sent to a TCM decoder 18. The TCM decoder 18 outputs soft information regarding the symbols which can be thresholded to produce hard-decision decoded symbols. Since the operation of the TCM encoder 14 can be modeled as an FSM, the decoding function of the TCM

decoder 18 can be based on the reduced-state soft-input/soft-output (RS-SISO) algorithm according to the invention as discussed below.

Please replace the paragraph at p. 4, line 28 through p. 5, line 10 with the following amended paragraph:

Figure 2 shows a section of a trellis describing an FSM which models a process such as TCM encoding, ISI interference, and others. Here, the trellis is a state diagram with explicit denotation of index in time, space, or some other dimension. Generally, such a trellis consists of a set of states $\mathbb{S} = \{S_1, S_2, \ldots, S_N\}$. The state of the FSM at time k is $s_k = S_i \in \mathbb{S}$. The trellis transition t is deterministically driven by a data source that outputs sequence $\{a_1, a_2, \ldots, a_K\}$ with symbol a_k drawn from the M-ary ~~closed~~ set of symbols $A = \{A_0, A_1, \ldots, A_{M-1}\}$. Therefore, in an FSM, each transition t is associated with a starting state $s^s(t)$, an ending state $s^e(t)$, an input symbol $a(t)$ and a output symbol $x(t)$. For simplicity of presentation, only the FSM whose state is defined as $s_k = (a_{k-L}, a_{k-L+1}, \ldots, a_{k-1})$ is discussed here. However, one skilled in the art would realize that such a restriction is not necessary to the functionality of the FSM and related operations. An FSM whose state is defined as $s_k = (a_{k-L}, a_{k-L+1}, \ldots, a_{k-1})$ is said to have memory length L and the number of states $N = M^L$. Consequently, for a transition $t_k =,(a_{k-L}, \ldots, a_k)$, the relevant expressions are: $s^S(t_k) = (a_{k-L}, \ldots, a_{k-1})$, $S^e(t_k) = (a_{k-L+1}, \ldots, a_k)$, $a(t_k) = a_k$, and the output symbol $x_k = x(t_k)$. The function $x(\cdot)$ can be any mapping, such as a 1-to-1 mapping, an n-to-1 mapping, an n-to-m mapping, or some other mapping.

Please replace the paragraph at p. 5, lines 11-15 with the following amended paragraph:

For clarity of illustration, Figure 2 shows a section of a trellis for a relatively simple FSM with $N = 4$ and $M = 2$. Thus, the trellis consists of a set of states $\mathbb{S} = \{0, 1, 2, 3\}$. The state of the FSM at time k is $s_k = S_i \in \mathbb{S}$. The trellis transition t is deterministically driven by a data source that outputs sequence $\{a_1, a_2, \ldots, a_K\}$ with symbol a_k drawn from the 2-ary ~~closed~~ set of symbols $A = \{0, 1\}$. In other words, the symbols are binary.

Please replace the paragraph at p. 7, lines 5-10 with the following amended paragraph:

Note that each element of $\tilde{\mathbf{a}}_k^f(i)$ and $\tilde{\mathbf{a}}_k^b(i)$ is defined on the M-ary ~~closed~~ set of symbols $A = \{A_0, A_1, \ldots, A_{M-1}\}$. Alternatively, each element of $\tilde{\mathbf{a}}_k^f(i)$ and $\tilde{\mathbf{a}}_k^b(i)$ can be defined on some set partitioning of the M-ary ~~closed~~ set of symbols $A = \{A_0, A_1, \ldots, A_{M-1}\}$. For example, if $M =$

8 and $\boldsymbol{A} = \{A_0, A_1, \ldots, A_7\}$, elements of $\tilde{\mathbf{a}}_k^f(i)$ and $\tilde{\mathbf{a}}_k^b(i)$ may be defined on the set partitioning $\boldsymbol{D} = \{D_0, D_1\}$, where D_0 encompasses $\{A_0, A_1, \ldots, A_3\}$ and D_1 encompasses $\{A_4, A_5, \ldots, A_7\}$. Other types of set partitioning are also possible.

Please replace the paragraph at p. 8, line 30 through p. 9, line 4 with the following amended paragraph:

Note that each element of $\tilde{\mathbf{a}}_k^f(i)$ and $\tilde{\mathbf{a}}_k^b(i)$ is defined on the M-ary ~~closed~~ set of symbols $\boldsymbol{A} = \{A_0, A_1, \ldots, A_{M-1}\}$. Alternatively, each element of $\tilde{\mathbf{a}}_k^f(i)$ and $\tilde{\mathbf{a}}_k^b(i)$ can be defined on some set partitioning of the M-ary ~~closed~~ set of symbols $\boldsymbol{A} = \{A_0, A_1, \ldots, A_{M-1}\}$. For example, if M = 8 and $\boldsymbol{A} = \{A_0, A_1, \ldots, A_7\}$, elements of $\tilde{\mathbf{a}}_k^f(i)$ and $\tilde{\mathbf{a}}_k^b(i)$ may be defined on the set partitioning $\boldsymbol{D} = \{D_0, D_1\}$, where D_0 encompasses $\{A_0, A_1, \ldots, A_3\}$ and D_1 encompasses $\{A_4, A_5, \ldots, A_7\}$. Other types of set partitioning are also possible.